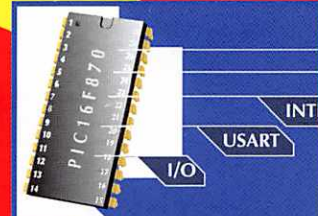
...e costruisci il tuo LABORATORIO DIGITALE

6,90 €

55

HARDWARE

DIGITALE DI BASE

MICROCONTROLLER

DIGITALE AVANZATO

TOTALMENTE PROGRAMMABILE!!!

9 771824 363008 41055

Peruzzo & C.

Direttore responsabile:
ALBERTO PERUZZO
Direttore Grandi Opere:
GIORGIO VERCELLINI
Consulenza tecnica e traduzioni:
CONSULCOMP S.n.c.
Pianificazione tecnica
LEONARDO PITTON

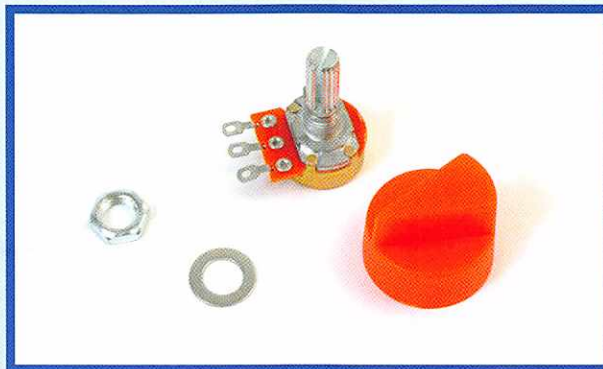
Direzione, Redazione, Amministrazione: viale Ercole Marelli 165, Tel. 02/242021, 20099 Sesto San Giovanni (Mi). Pubblicazione settimanale. Registrazione del Tribunale di Monza n. 1738 del 26/05/2004. Spedizione in abbonamento postale gr. II/70; autorizzazione delle Poste di Milano n. 163464 del 13/2/1963. Stampa: Grafiche Porpora s.r.l., Cernusco S/N (MI). Distribuzione SO.DI.P. S.p.A., Cinisello Balsamo (MI).

"ELETTRONICA DIGITALE" si compone di 70 fascicoli settimanali da suddividere in 2 raccoglitori.

RICHIESTA DI NUMERI ARRETRATI. Per ulteriori informazioni, telefonare dal lunedì al venerdì ore 9.30-12.30 all'ufficio arretrati tel. 02/242021. Se vi mancano dei fascicoli o dei raccoglitori per completare l'opera, e non li trovate presso il vostro edicolante, potrete riceverli a domicilio rivolgendovi direttamente alla casa editrice. Basterà compilare e spedire un bollettino di conto corrente postale a PERUZZO & C. s.r.l., Ufficio Arretrati, viale Marelli 165, 20099 Sesto San Giovanni (MI). Il nostro numero di c/c postale è 42980201. L'importo da versare sarà pari al prezzo dei fascicoli o dei raccoglitori richiesti, più le spese di spedizione € 3,10 per pacco. Qualora il numero dei fascicoli o dei raccoglitori sia tale da superare il prezzo globale di € 25,82 e non superiore a € 51,65, l'invio avverrà per pacco assicurato e le spese di spedizione ammonteranno a € 6,20. La spesa sarà di € 9,81 da € 51,65 a € 103,29; di € 12,39 da € 103,29 a € 154,94; di € 14,98 da € 154,94 a € 206,58; di € 16,53 da € 206,58 in su. Attenzione: ai fascicoli arretrati, trascorse dodici settimane dalla loro distribuzione in edicola, viene applicato un sovrapprezzo di € 0,52, che andrà pertanto aggiunto all'importo da pagare. Non vengono effettuate spedizioni contrassegno. Gli arretrati di fascicoli e raccoglitori saranno disponibili per un anno dal completamento dell'opera. IMPORTANTE: è assolutamente necessario specificare sul bollettino di c/c postale, nello spazio riservato alla causale del versamento, il titolo dell'opera nonché il numero dei fascicoli e dei raccoglitori che volete ricevere.

IN REGALO in questo fascicolo

1 Altoparlante da 8 Ω

IN REGALO nel prossimo fascicolo

1 Potenziometro da 50 K Log. con asse, completo di dado e rondella
1 Manopola per il potenziometro

COME RACCOGLIERE E SUDDIVIDERE L'OPERA NELLE 4 SEZIONI

L'Opera è composta da 4 sezioni identificabili dalle fasce colorate, come indicato sotto. Le schede di ciascun fascicolo andranno suddivise nelle sezioni indicate e raccolte nell'apposito raccoglitore, che troverai presto in edicola. Per il momento, ti consigliamo di suddividere le sezioni in altrettante cartellette, in attesa di poterle collocare nel raccoglitore. A prima vista, alcuni numeri di pagina ti potranno sembrare ripetuti o sbagliati. Non è così: ciascuno fa parte di sezioni differenti e rispecchia l'ordine secondo cui raccogliere le schede. **Per eventuali domande di tipo tecnico scrivere al seguente indirizzo e-mail:** elettronicadigitale@microrobots.it

Hardware Montaggio e prove del laboratorio

Digitale di base Esercizi con i circuiti digitali

Digitale avanzato Esercizi con i circuiti sequenziali

Microcontroller Esercizi con i microcontroller

L'altoparlante

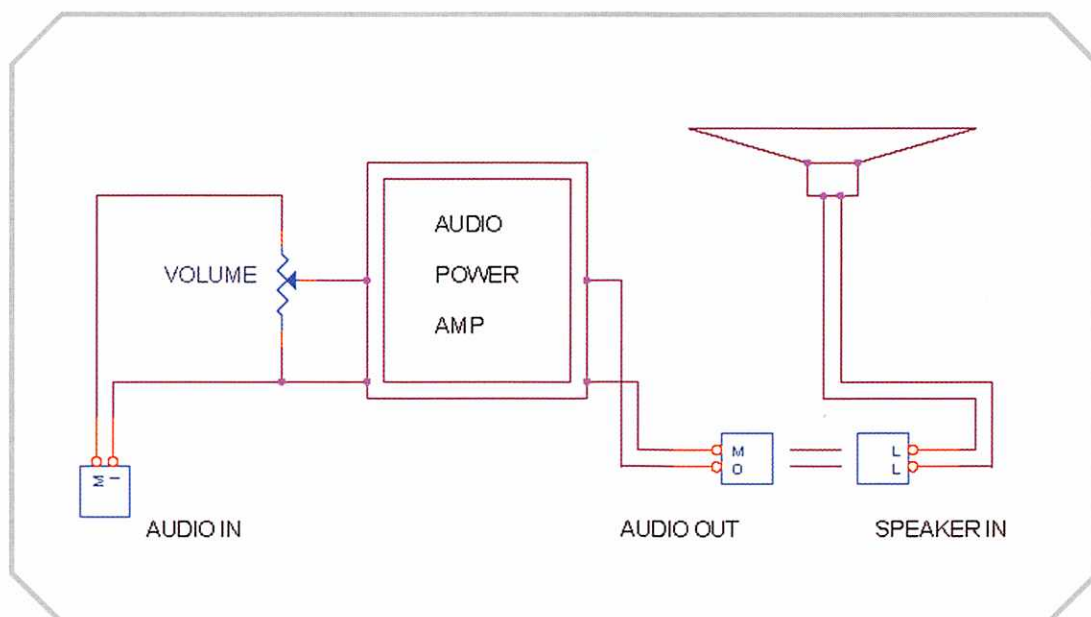
L'altoparlante da 2 pollici.

L'altoparlante si monta in questa zona.

Con questo fascicolo viene fornito l'altoparlante.
Se lo installiamo ora, sarà pronto per l'utilizzo.

Questo altoparlante si monta su una delle due sedi con griglia all'esterno del pannello superiore. Il suo suono migliorerà montando il coperchio, specialmente per le frequenze più basse. Si tratta di un piccolo altoparlante da 8 Ω di impedenza, 2 pollici di diametro circa 5 cm, e ha una potenza di circa 0,5 Watt.

Fissaggio

L'altoparlante si monta inserendolo in una sede circolare in cui rimarrà praticamente incastrato. Dato che potrebbero esserci piccole differenze di misura da un altoparlante all'altro, se l'inserimento viene fatto con precisione, l'altoparlante dovrà rimanere fissato da solo. Dobbiamo ruotarlo fino a quando le sue connessioni rimarranno rivolte verso la parte inferiore, per facilitarne i collegamenti. L'installazione definitiva si può eseguire utilizzando 3 o 4 gocce di colla specifica, oppure piegando con il saldatore parte del bordo rotondo di plastica che circonda l'altoparlante.

Schema elettrico, collegamento dell'altoparlante.

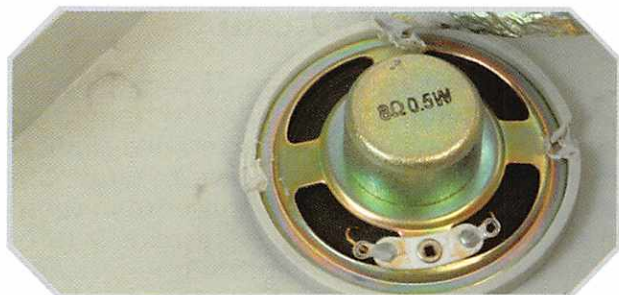
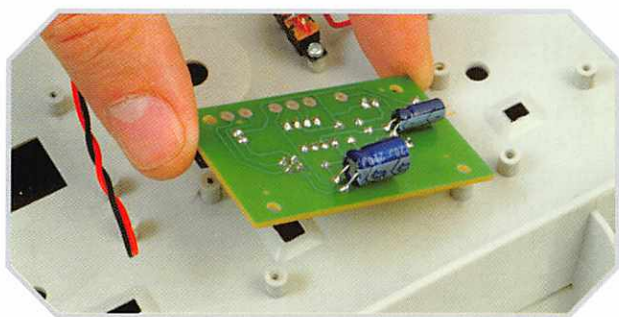
Sede circolare dell'altoparlante.

L'altoparlante si deve inserire nel suo alloggiamento.

Se si sceglie quest'ultima opzione non bisogna applicare direttamente la punta del saldatore sull'altoparlante, né al bordo di plastica, altrimenti si sporcherà e avrà un cattivo odore di plastica fusa. Vi consigliamo di utilizzare il seguente trucco: fasciate la punta del saldatore freddo – per non bruciarvi – con un paio di giri di alluminio da cucina, dopodiché riscaldate il saldatore e in questo modo trasmetterete calore ma senza sporcare la punta del saldatore stesso. Infatti la plastica fusa entrerà in contatto con il foglio di alluminio che potremo gettare dopo aver terminato l'operazione.

Collegamento

Il collegamento dell'altoparlante si esegue con due pezzi di filo, uno di colore rosso e l'altro di colore nero, già preventivamente forniti, devono essere intrecciati fra di loro e collegati nel modo illustrato nelle fotografie: uno dei capi di ogni filo va all'altoparlante e gli altri due estremi ai punti di saldatura identificati sulla scheda audio DG15 come L1 e L2. In questo modo i collegamenti dell'altoparlante saranno disponibili sul connettore identificato come SPEAKER IN, sul pannello frontale.

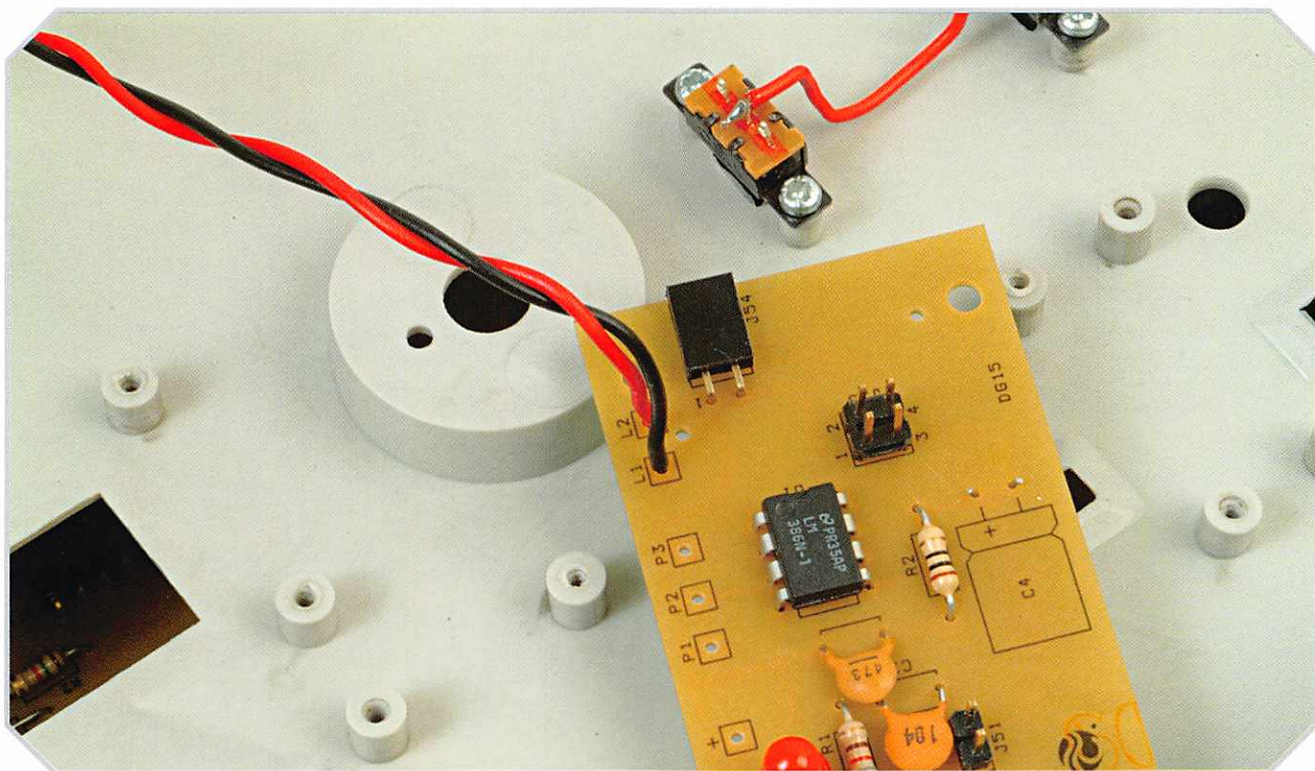
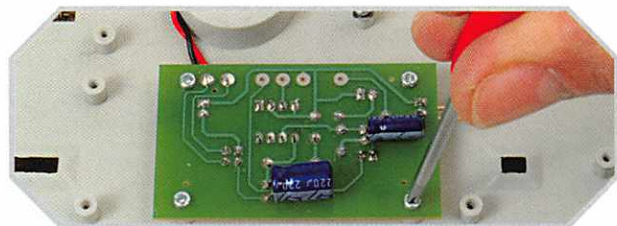
Collegamento dell'altoparlante.

Fissaggio mediante calore.

È consigliabile togliere la scheda audio.

Collegamenti alla scheda DG15

I collegamenti esterni dell'altoparlante sono situati sulla scheda DG15 e sono accessibili dal pannello frontale.

Per realizzare il collegamento a questa scheda sono necessari due pezzi di filo, uno di colore rosso e l'altro nero, entrambi da 13 cm di lunghezza; è sufficiente spelare per circa 3 mm i fili da entrambi i lati e saldarli sulle piazzole di saldatura indicate come L1 e L2. Vi consigliamo di saldare il filo nero a L1 e il filo rosso a L2. Questa operazione si potrebbe eseguire senza togliere la scheda DG15, ma è meglio smontarla togliendo le quattro viti e dopo averla saldata installarla nuovamente. Anche questa volta non stringeremo le viti, dato che sarà necessario toglierla nuovamente per collegare il potenziometro del volume.

Collegamenti interni

L'altoparlante si può collegare all'uscita dell'amplificatore mediante due ponticelli, inseriti in senso orizzontale sul connettore indicato come AUDIO OUT e SPEAKER IN.

Collegamenti alle piazzole L1 (nero) e L2 (rosso).

Scheda audio già collegata all'altoparlante.

Collegamento dell'altoparlante all'esterno.

Collegamenti esterni

È anche possibile utilizzare l'altoparlante in modo indipendente per pilotarlo tramite un altro circuito, collegandosi ai terminali indicati come SPEAKER IN con un cavetto terminato su due connettori a due vie, inserito in senso verticale, come possiamo osservare nella fotografia. Questo collegamento si può fare arrivare a due dei terminali siglati da 1 a 16, in questo modo sarà possibile avere a disposizione il collegamento dell'altoparlante sulle molle del laboratorio.

Utilizzo

È già possibile utilizzare in modo indipendente l'altoparlante, invece l'amplificatore audio non potrà essere impiegato fino al prossimo fascicolo con cui verrà fornito il potenziometro del volume.

Vista generale del laboratorio.

Generatore audio

Il timer 555 si può utilizzare per costruire un oscillatore che lavori su frequenze comprese all'interno della banda audio.

Il circuito

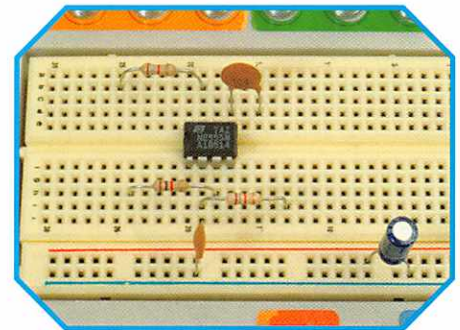
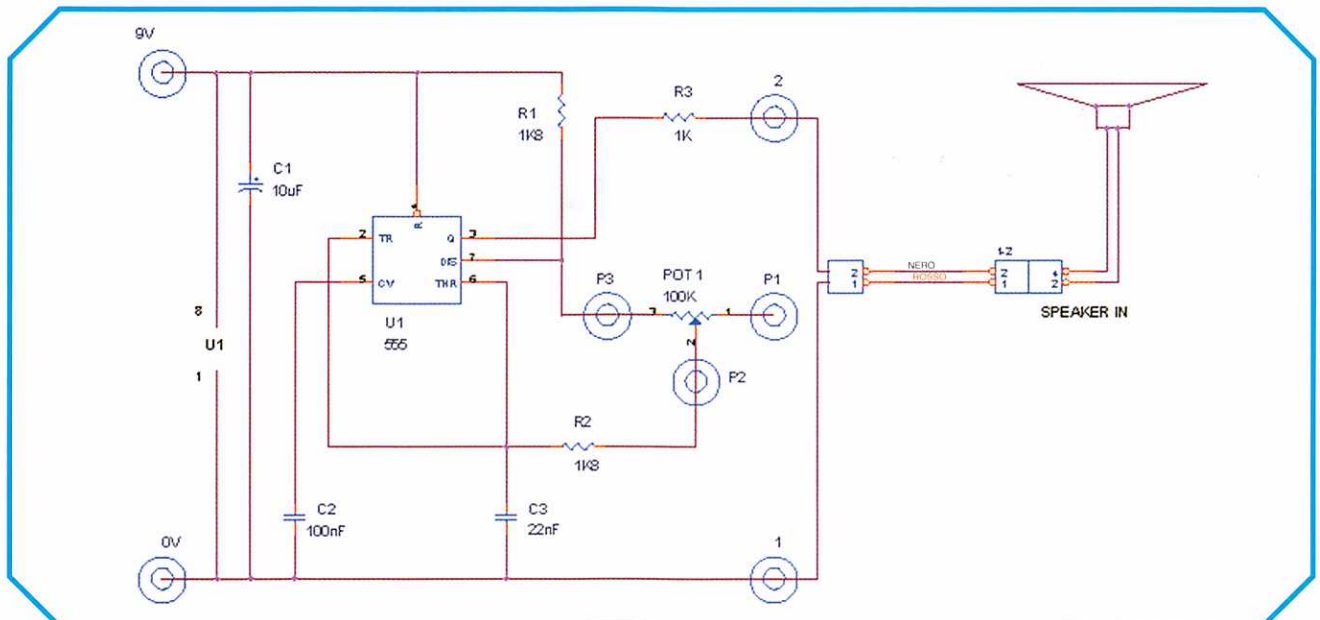
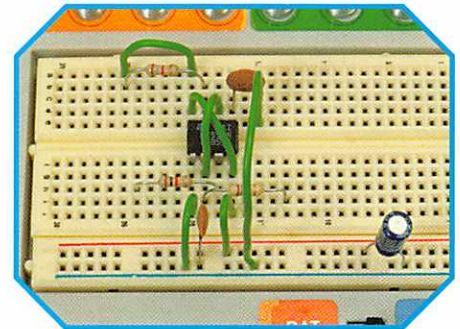
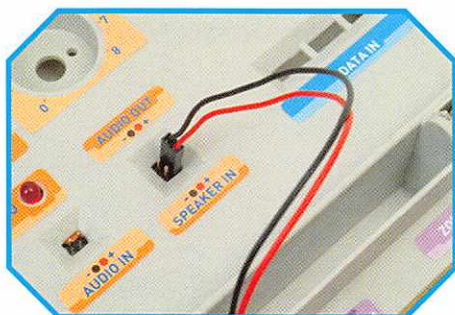
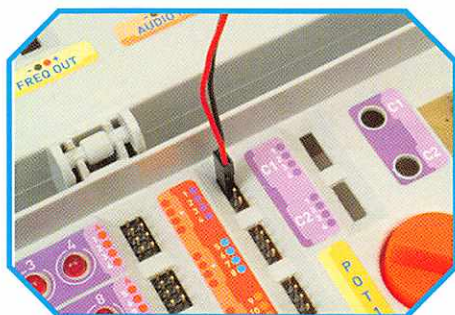
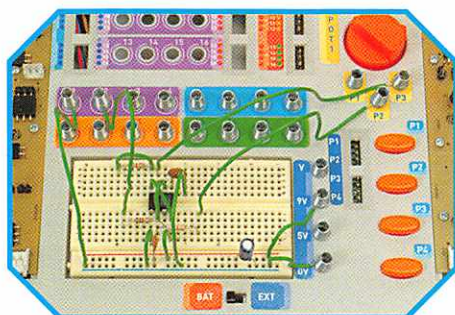
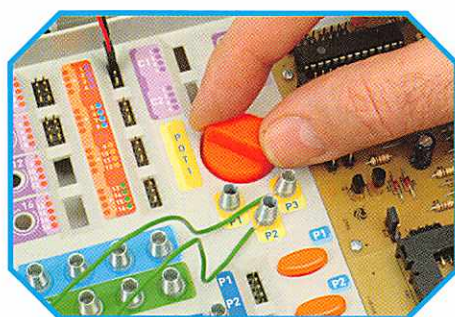
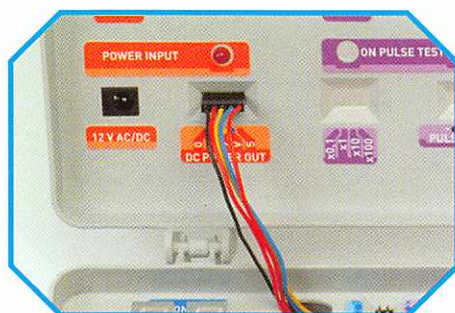
Il circuito è molto conosciuto in quanto consigliato dai costruttori del 555 per costruire oscillatori astabili. Una delle resistenze che determina la frequenza è R1, mentre l'altra è formata dalla somma delle resistenze R2 e dal valore che in quel momento è determinato dalla posizione del potenziometro POT1.

La formula che utilizzeremo per determinare la frequenza del segnale di uscita è:

$$f = 1{,}44/(R1 + 2R2 + 2POT1)C$$

Per poterla applicare correttamente la resistenza deve essere espressa in Ohm e la capacità in Farad, in modo che il valore della frequenza ottenuta sia in Hertz. Ricordiamo che l'altro componente che determina la frequenza è il condensatore C2. Se aumentiamo il suo valore la frequenza si abbassa.

Con i valori utilizzati nello schema si otten-

Componenti sulla scheda Bread Board.

Cablaggio della scheda.

Schema del circuito.

Collegamenti del cavetto su SPEAKER IN.

L'altro capo del cavetto si collega a 1-2.

Cablaggio esterno alla scheda Bread Board.

Con il potenziometro si modifica la frequenza di uscita.

Collegamento della alimentazione.

gono frequenze fra 350 e 12.000 Hz sfruttando tutta la corsa del potenziometro. Ruotando il comando di POT1 è possibile ottenere qualsiasi frequenza compresa fra queste due, tutte queste sono comprese nella banda udibile. Sull'uscita del circuito inseriamo una resistenza R3 per limitare la corrente di uscita dello stesso, inviando così un segnale all'altoparlante sufficiente per verificare il funzionamento del circuito, mantenendo un livello acustico basso. Tenete presente che non stiamo adoperando nessun amplificatore, se lo desiderate potete ridurre il valore della resistenza R3, ma in nessun caso dovrà essere minore di 100 Ω.

Montaggio

Il montaggio dei componenti sulla scheda si esegue come d'abitudine, tenendo presente l'orientamento del circuito integrato e quello del condensatore C1 che è elettrolitico. Il collegamento all'altoparlante si esegue mediante un cavetto a due connettori collegato a SPEAKER IN, portando l'altro estremo ai connettori ausiliari delle molle, ad esempio 1-2, in modo da avere a disposizione i collegamenti dell'altoparlante sulle molle 1 e 2. Dovremo collegare l'alimentazione sui terminali 1 e 8 dell'integrato e l'alimentazione generale potrà essere tra 5 o 9 Volt indistintamente.

L'esperimento

Dopo aver verificato il lavoro di montaggio posizioniamo il comando del potenziometro POT1 all'incirca a metà della sua corsa e colleghiamo l'alimentazione, a questo punto si deve udire un segnale continuo sull'altoparlante, ruotando il comando del potenziometro potremo sentire come cambia la frequenza del segnale di uscita.

LISTA DEI COMPONENTI

U1	Circuito integrato 555
R1, R2	Resistenza 1K8 (marrone, grigio, rosso)
R3	Resistenza 1 K (marrone, nero, rosso)
C1	Condensatore 10 µF elettrolitico
C2	Condensatore 100 nF
C3	Condensatore 22 nF

Esercizio 15: la USART, compilazione, simulazione e sviluppo

Nel fascicolo precedente abbiamo imparato a progettare un programma che stabilisce una comunicazione seriale tra il PC e il PIC tramite il modulo USART di quest'ultimo. Compileremo il programma, lo simuleremo e per i più esperti indicheremo come implementarlo fisicamente e come interagire con il PC.

Compilazione

Per verificare se il programma preparato non ha errori dobbiamo aprire MPLAB, creare un progetto e nella finestra di edizione di quest'ultimo, aggiungere il file da compilare. Fatto questo selezioneremo Build All per assemblare e compilare il codice.

Il codice compila senza errori e presenta solamente dei messaggi riguardanti la corrispondenza dei registri con i banchi di memoria.

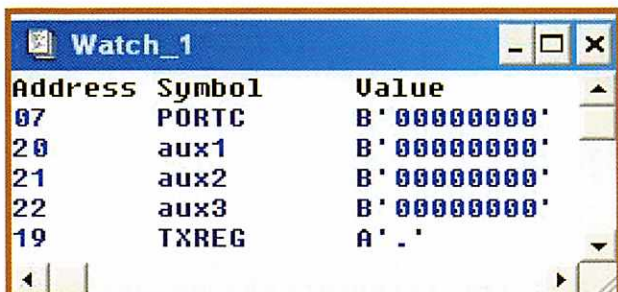
Simulazione

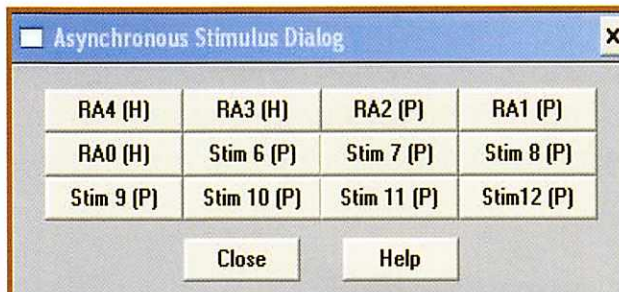
Per simulare l'esercizio con MPLAB apriremo la finestra dei Registri delle Funzioni Speciali e una finestra con i registri che ci interessa vedere in modo indipendente. Nel nostro caso possono essere: PORTC, aux1, aux2, aux3 e TXREG. La visualizzazione dei dati è consigliabile farla in binario per i primi quattro e in ASCII per l'ultimo. In questo modo potremo vedere più chiaramente se il programma risponde alle nostre aspettative. Nell'immagine presente nella pagina successiva, possiamo vedere come si presenta questa finestra dopo essere stata configurata.

Se cominciamo l'esecuzione del programma passo a passo, vedremo che questo viene eseguito normalmente fino a quando arriva alla parte di codice che deve inviare i caratteri ed entra nella subroutine di invio. La simulazione si fermerà in un ciclo di attesa fino a quando non terminerà la trasmissione. Per poter uscire da questo ciclo, forzeremo un valore sul registro TXSTA utilizzando la finestra Modify. Nell'immagine in figura è riportato un esempio di come forzare un valore all'interno del registro.

Se forziamo il valore del registro TXSTA a FF (in binario '11111111'), ogni volta che entra nella subroutine di trasmissione uscirà da questa senza problemi.

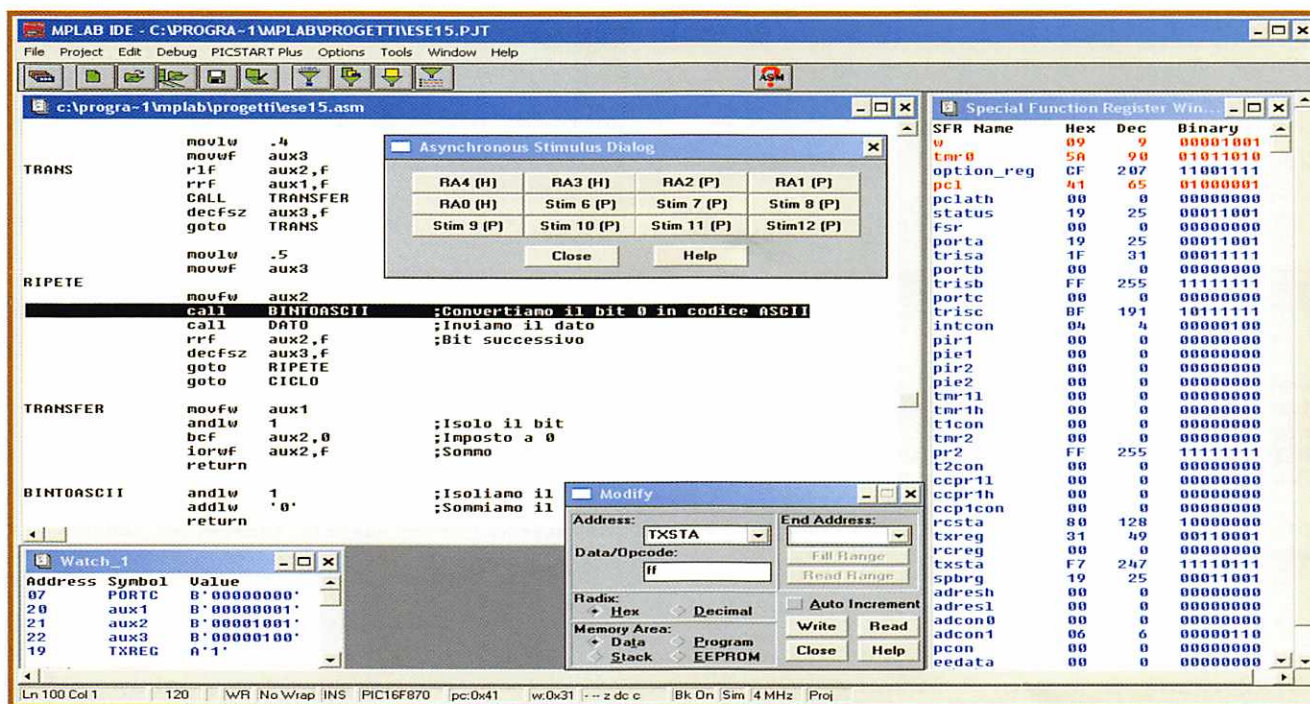
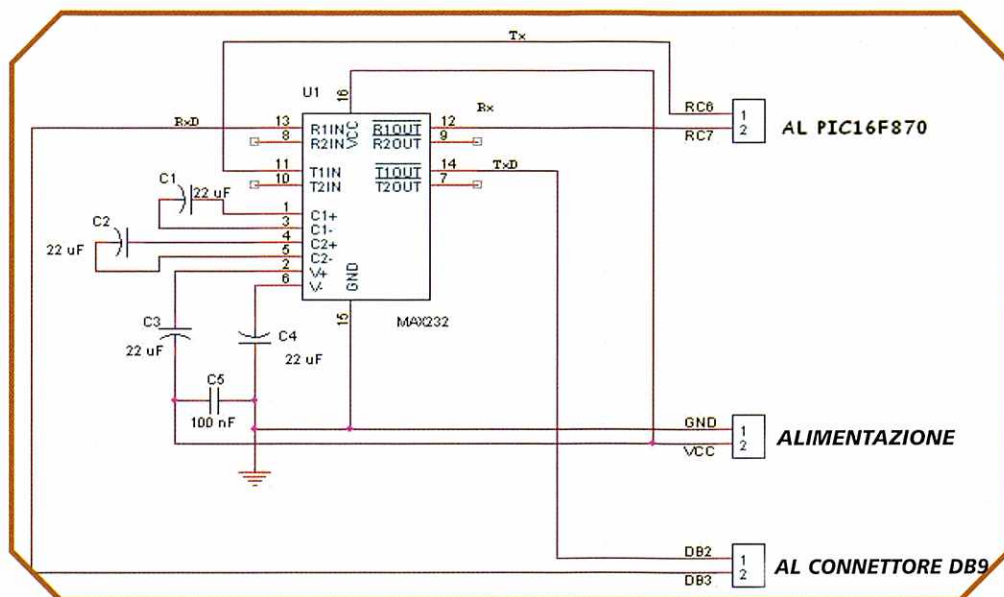
Continuando a eseguire il programma vedremo che, dopo aver letto il valore della Porta A, viene invertito l'ordine dei bit, di conseguenza il valore della variabile aux3 cambia, fino a ottenere l'inversione completa del valore letto sulla porta; successivamente si trasmette il valore ASCII ottenuto dalla combinazione dei pin di ingresso. Se non simuliamo nessun valore di ingresso verranno trasmessi cinque '0' e non potremo vedere come viene riordinato il valore da trasmettere, ma se nella esecuzione successiva, prima di acquisire il valore della porta, simuleremo degli stimoli sui pin, potremo vedere tutti i passi eseguiti in modo corretto.

Risultato della compilazione.

Finestra dove si vede il valore dei registri più interessanti.

Forziamo il valore del registro TXSTA per uscire dal ciclo.

Con il simulatore di stimoli asincroni forziamo il valore di ingresso.

Con il simulatore di stimoli asincroni simuleremo un ingresso qualsiasi, ad esempio quello riportato nella figura. Ricordate che per attivare il livello alto sui pin selezionati, dovremo cliccare con il pulsante sinistro del mouse sul pulsante corrispondente al pin, uno per uno. Attiveremo un pin, eseguiremo una linea di programma, attiveremo un altro pin eseguiremo un'altra linea, e così via. Verificheremo che in queste condizioni, al momento della simulazione, il programma risponde correttamente.

Implementazione e sviluppo

Vediamo ora come si implementa questo esercizio fisicamente, per quei lettori che vorran-

Aspetto di MPLAB durante la simulazione.

Schema elettrico della comunicazione via seriale RS232.

UNITÀ	COMPONENTE
1	Circuito integrato RS232 (p.e. MAX232)
4	Condensatori da 22 µF
1	Condensatore da 100 nF
1	Connettore DB9 femmina

Componenti necessari per il montaggio del circuito elettrico di comunicazione RS232.

no "andare più avanti", e desiderano svilupparlo correttamente.

Per stabilire una comunicazione seriale con il modulo USART è necessario disporre di un circuito esterno al microcontroller che implementi il protocollo di comunicazione. Il protocollo che si utilizza in questi casi è il RS232.

Circuito di ingresso

Dobbiamo usare cinque interruttori per gli ingressi. Nel caso non avessimo a disposizione gli interruttori necessari potremo eseguire un semplice montaggio basato sulle resistenze, come abbiamo visto negli esercizi precedenti.

Montaggio necessario per stabilire la comunicazione

Dobbiamo comunicare usando il protocollo RS232, quindi è necessario eseguire un montaggio specifico per questa comunicazione. Un circuito elettrico tipico per comunicazioni RS232 è quello riportato nella figura. Come potete verificare, abbiamo specificato nel circuito con quali terminali del PIC si collega (RC6 e RC7), come si collega l'alimentazione e a quali pin del connettore DB9 deve essere unita la linea di comunicazione.

Tramite il terminale RC6 del PIC uscirà il dato da trasmettere, che entrerà nel circuito integrato tramite il terminale 11 (T1IN). Il dato in formato RS232, uscirà dall'integrato tramite il terminale 14 (T1OUT) verso il connettore DB9, sul suo terminale DB2. Quando il PC invia un dato, questo arriva sul circuito tramite il pin DB3 del connettore e arriva al circuito integrato tramite il terminale 13 (R1IN). L'integrato converte il dato per poterlo trasmettere al PIC, facendolo uscire tramite il pin 12 (R1OUT) per arrivare a questo tramite il terminale RC7.

Avendo chiari questi collegamenti, è sufficiente inserire i condensatori indicati nello schema, alimentare il circuito integrato e unire il terminale 5 (DB5) del connettore DB9 a massa.

Hyper Terminal di Windows

Parliamo ora di questo programma, anche se non è l'unico software di comunicazione con il quale può funzionare la nostra applicazione. Questo programma si trova nel menù Comunicazioni, all'interno di Accessori, nel menù a tendina di Programmi. Facendo il dop-

Apriamo Hyper Terminal e creiamo una connessione.

pio clic sull'icona del programma apparirà una videata per creare una nuova comunicazione. In questa finestra dobbiamo dare un nome al collegamento che vogliamo creare e possiamo scegliere l'icona che ci risulta più appropriata. Fatto questo, la videata successiva serve per stabilire il dispositivo tramite il quale si stabilirà la comunicazione: modem, sockets o porta. Nel nostro caso sceglieremo la porta seriale COM1, dove collegheremo il nostro cavo di comunicazione. Dopo aver scelto il dispositivo dovremo configurarlo, e questo lo si fa mediante una nuova finestra. Nella finestra Proprietà di COM1, dobbiamo configurare i parametri della nostra comunicazione, che nel nostro caso saranno 9.600 bit per secondo, 8 bit di dati, parità pari, un bit di stop e controllo di flusso di tipo hardware. Potete provare a cambiare alcuni di questi parametri, osservando che cosa succede nella comunicazione. Configurata la comunicazione si apre una finestra di lavoro in cui verranno visualizzati i messaggi inviati tramite il PIC.

Avvertenze

I componenti per montare il circuito di comunicazione seriale 232 non sono forniti con l'opera, quindi chi fosse interessato alla realizzazione di questo montaggio, li dovrà acquistare in un qualsiasi negozio di componenti elettronici.

In ogni caso l'obiettivo principale era l'acquisizione delle conoscenze necessarie per lavorare con qualsiasi dei dispositivi del PIC, quindi per quanto riguarda la USART consideriamo raggiunto il nostro obiettivo.

Connetti a

usart

Immettere i dettagli per il numero telefonico da comporre:

Paese: Italia (39)

Indicativo località:

Numero di telefono:

Connetti: COM1

OK Annulla

Dispositivo tramite il quale si effettua la comunicazione.

Proprietà - COM1

Impostazioni della porta

Bit per secondo: 9600

Bit di dati: 8

Parità: Nessuno

Bit di stop: 1

Controllo di flusso: Hardware

Ripristina

OK Annulla Applica

Configurazione dei parametri della comunicazione.

Esercizio: controllo di un forno

Presentiamo ora un progetto reale in cui sono utilizzati molti dei dispositivi del PIC16F870, come ad esempio le memorie FLASH e EEPROM interne, il convertitore A/D, interrupt, timer e le porte di comunicazione seriale (USART). Utilizzando i microcontroller potremo risolvere e controllare una moltitudine di processi industriali in modo semplice, pratico ed economico.

Controllo della temperatura per il modellamento dei pezzi

Vogliamo realizzare il controllo di un processo in cui si riscaldano i pezzi in un forno per il loro successivo modellamento.

Lavoreremo con due tipi di pezzi:

– Pezzi A: questi pezzi potranno essere modellati se rimarranno nel forno per 5 secondi a una temperatura superiore ai 125° (Forno 75-125.asm).

– Pezzi B: questi pezzi potranno essere modellati se rimarranno nel forno per 5 secondi a una temperatura superiore a 150° (Forno 100-150.asm).

In base ai pezzi che si trovano sulla catena dobbiamo eseguire un controllo oppure l'altro.

Per i pezzi A il nostro forno ci avviserà che ci stiamo avvicinando alla temperatura specificata (125°) dopo aver superato i 75° C. In questo modo l'addetto potrà essere avvisato e preparare quanto serve per il processo successivo.

L'allarme sparisce quando la temperatura non è superiore ai 75°. Se al contrario la temperatura ha raggiunto quella specificata, sparisce l'allarme e viene indicato che la temperatura è quella corretta. Se questo stato si mantiene per 5 secondi il pezzo verrà considerato riscaldato e si introdurrà all'interno del forno il pezzo successivo.

Per i pezzi B il processo è identico ma cambiano le temperature. Essendo di materiale diverso, per il loro successivo modellamento, hanno bisogno di rimanere nel forno per 5 secondi a temperatura superiore a 150°, generando l'allarme di avviso all'addetto quando la temperatura del forno supera i 100°.

Dispositivi utilizzati dal PIC.

Esecuzione del progetto

Per la realizzazione di questo progetto abbiamo a disposizione una sonda attiva di temperatura all'interno del forno. Il forno avrà un range di temperatura fra 0° e 255° C, e l'uscita proporzionale che otterremo dalla sonda sarà compresa tra 0 e 5 Volt.

Questa sonda potrà essere simulata con un potenziometro.

L'uscita di questa sonda sarà interpretata dal microcontroller che convertirà questo valore in binario, per poterlo elaborare (convertitore A/D). Ogni volta che eseguiamo una conversione genereremo un interrupt in cui verificheremo il valore ottenuto.

Se il valore è compreso all'interno del range di allarme, indicheremo questa situazione facendo lampeggiare un diodo LED. Otterremo questo utilizzando il Timer0. Se il valore è compreso all'interno del range corretto di riscaldamento del pezzo, utilizzeremo lo stesso indicatore ma questa volta fornendo una luce fissa.

Verificheremo che il pezzo rimanga in quella situazione per 5 secondi e dopo aver soddisfatto questa condizione inseriremo un nuovo pezzo nel forno. Il movimento del nastro trasportatore lo indicheremo con un altro diodo

Processo industriale di modellamento dei pezzi.

Visualizzazione sul PC delle variabili da controllare.

LED e sarà prodotto da un motore collegato indirettamente (tramite una circuiteria addizionale) a una uscita del microcontroller.

Per simulare il montaggio collegheremo solamente il LED all'uscita.

Dovremo inoltre indicare il numero di pezzi eseguiti e la temperatura di lavoro attuale del forno. Mediante la comunicazione seriale del microcontroller (modulo USART) invieremo al terminale di un PC questi valori. In questo modo potremo osservare il funzionamento del processo in tempo reale tramite una postazione remota.

Utilizzando il Bootloader, file che si trova nelle directory (Bootloader.ascu) che porta questo nome inserita nel primo CD-R, potremo, tramite lo stesso programma e utilizzando lo stesso tipo di comunicazione, selezionare il processo da realizzare. Potremo scrivere il nuovo programma da eseguire per il microcontroller, in base ai pezzi che si desidera automatizzare. Fermando il processo e facendolo ripartire tenendo premuto il reset, il microcontroller si ferma e attende che venga caricato un nuovo programma, spegnendo e alimentando nuovamente il circuito manderemo in funzione il programma dell'utente per il processo scelto.

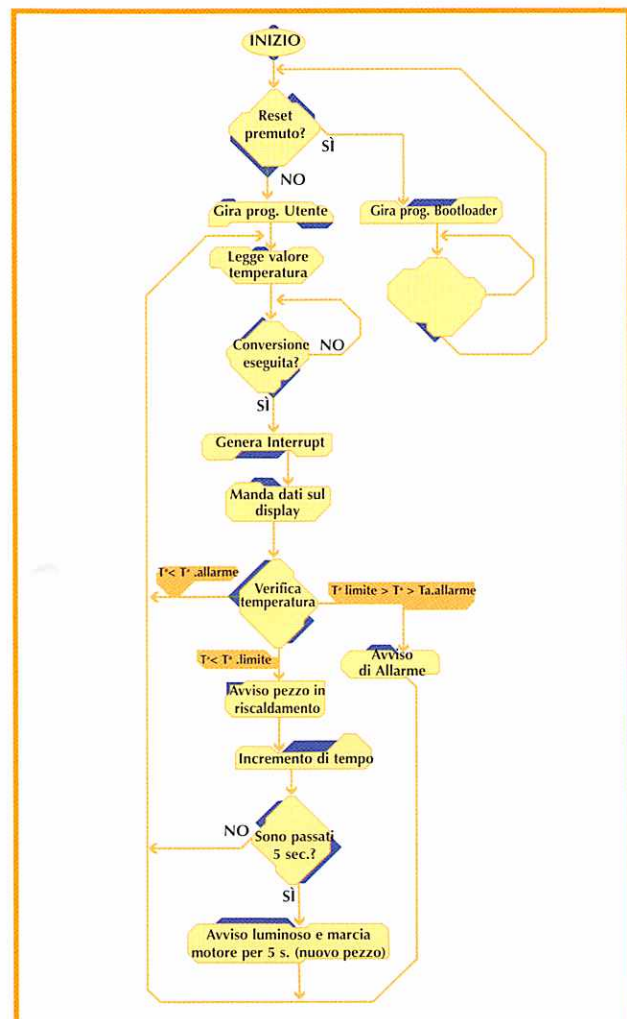
Il Bootloader e le relative modifiche ai programmi per poter lavorare con esso, verranno presentate nei prossimi fascicoli.

Organigramma

Mediante un organigramma pianificheremo una soluzione generale al progetto e partendo da essa svilupperemo il codice più facilmente.

Codice

Per spiegare come è stata risolta l'applicazione ci concentreremo su uno dei programmi, dato che l'altro è molto simile, con l'unica differenza dei valori di temperatura. Risolveremo quindi il programma per i pezzi A, e spiegheremo i cambiamenti da eseguire, partendo da questo, per fare il programma corrispondente ai pezzi di tipo B.

Il codice inizia con i commenti relativi al programma, le definizioni del PIC e la libreria dei registri. Possiamo definire le costanti per i valori di temperatura. Così 125 sarà '01111101' in binario e 75 sarà '01001011'. Seguiranno la disposizione per gli indirizzi della memoria di programma e il vector di interrupt mediante le direttive ORG.

Organigramma del progetto.

```
;       ESERCIZIO: Forno 75-125
;       -------------------------
;

        LIST P=16F870
        include "P16f870.INC"

volt3           EQU     b'01111101'     ;Raggiunti i 125° fissa l'uscita e inizia la sequenza
volt2           EQU     b'01001011'     ;Raggiunti i 75° ci avvisa della temperatura


;Inizio del programma
        org 0
        GOTO INIZIO

        org 4
        goto INT

        org 5
```

Intestazione del programma.

```
INIZIO  clrf    PORTA
        clrf    PORTB
        clrf    PORTC           ;Resettiamo la Porta C
        clrf    n_pezzi
        ;Configuriamo ADCON con: RC, canale2, Attivazione del "modulo del convertitore"
        ;Se non lo programmiamo con RC non lo potremo risvegliare dallo SLEEP!
        movlw   b'11000001'
        movwf   ADCON0
        ;Configuriamo ADCON con giustificazione a sinistra->lo carichiamo in ADRESH+tutto analogico;
        banksel ADCON1
        movlw   b'00001110'
        movwf   ADCON1
        clrf    TRISB           ;Configuriamo la Porta B come uscita digitale
        movlw   b'11001111'     ;Sul registro OPTION WDT=1/128
        movwf   OPTION_REG
        movlw   b'00000001'     ;Porta A: RA0 ingresso analogico, gli altri uscite
        movwf   TRISA
        movlw   b'10111111'
        movwf   TRISC           ;Impostiamo RC6/Tx come uscita e RC7/Rx come ingresso
        movlw   b'00100100'
        movwf   TXSTA           ;Attiviamo la trasmissione in modo asincrono, a 8 bit ad alta velocità
        movlw   .25
        movwf   SPBRG           ;Carichiamo il reg. generatore della velocità in baude in modo che si configuri con 9600
        banksel RCSTA
        bsf     RCSTA,SPEN      ;Abilitiamo la porta seriale
        movlw   .325000         ;Con questo valore otterremo i 5 sec.
        movwf   cont
        movwf   cont1

;*****Configurazione dell'interrupt per il salto quando si attiva la conversione
        banksel PIE1
        movlw   b'01000000'
        movwf   PIE1
        banksel INTCON
        movlw   b'11000000'
        movwf   INTCON
conver  bsf     ADCON0,GO
        sleep
        goto conver
```

Configurazione dei dispositivi.

```
;****************Routine di interrupt****************

        org 50
        banksel PORTB
INT     goto    CICLO
eti     btfsc   PORTB,5
        goto    temp
        movf    ADRESH,w
        sublw   volt3
        banksel STATUS
        btfss   STATUS,0        ;Se il carry è a zero:
        goto    ALLARME
        movlw   .325000         ;Con questo valore otterremo i 5 sec.
        movwf   cont
        movf    ADRESH,w
        sublw   volt2
        banksel STATUS
        btfss   STATUS,0        ;Se il carry è a zero:
        call    lampeg
        banksel PORTB
        clrf    PORTB
        banksel PIR1
REINIT  bcf     PIR1,6          ;Reinizializziamo il bit dell'interrupt
        retfie

;*********************************************************************************
;Subroutine che mantiene l'uscita RB4 costantemente accesa
ALLARME nop
        banksel PORTB
        bsf     PORTB,4
        decfsz  cont,F
        goto    eti1
        call    motore
temp    decfsz  cont1,F
        goto    eti1
        bcf     PORTB,5
eti1    nop
        banksel PIR1
        goto    REINIT
```

A partire dall'etichetta Inizio resetteremo le porte e configureremo i dispositivi, le tre porte, il CAD, il WDT, la USART e gli interrupt. Ci fermeremo in modalità SLEEP entrando in un ciclo fino a quando non si esegue la conversione. Definiremo tre nuove variabili all'inizio del programma.

Terminata la conversione si genera un interrupt e si esegue la routine dedicata a quest'ultima. In essa si analizza il valore acquisito, attivando l'uscita nel caso in cui la temperatura sia adeguata, avvertendo con il lampeggio dell'allarme il fun-

Routine di servizio dedicata all'interrupt.

```
;*****************************************************************
;Subroutine che provoca il lampeggio dell'uscita RB4

lampeg  clrf    PORTB
        bsf     STATUS,RP0      ;Passiamo al banco 1
        clrf    TRISB           ;Porta B uscite
        movlw   b'11010111'     ;divisore 256
        movwf   OPTION_REG
        bcf     STATUS,RP0      ;Torniamo al banco 0

LAMPEGG bcf     PORTB,4         ;Spegniamo il LED (RB4)
        call    RITARDO         ;Saltiamo alla routine di ritardo
        bsf     PORTB,4         ;Accendiamo il LED
        call    RITARDO
        return


RITARDO movlw   b'00000001'     ;Valore da contare 255
        movwf   TMR0            ;(complemento a 2)
        btfss   INTCON,T0IF
        goto    $-1             ;Va all'istruzione precedente
        bcf     INTCON,T0IF
        return

;*****************************************************************************
;Subroutine che mette in marcia il motore
motore  movlw   .325000
        movwf   cont
        movwf   cont1
        bsf     PORTB,5
        incf    n_pezzi,1
        return
;*****************************************************************************
;Subroutine della Usart: visualizziamo sul display la temperatura e il numero dei pezzi

VISUAL  bcf     PIR1,TXIF       ;Resettiamo il flag del trasmettitore
        movwf   TXREG           ;Scriviamo il dato
        bsf     STATUS,RP0      ;Passiamo al banco 1
CICLO1  btfss   TXSTA,TRMT      ;Verifichiamo se la trasmissione è terminata
        goto    CICLO1          ;Non è ancora terminata
        bcf     STATUS,RP0      ;Torniamo al banco 0
        return
```

Subroutine con funzioni diverse.

zionamento sull'uscita. Per semplificare il programma lo separeremo in pezzi di codice raggruppati in subroutine. Per il lampeggio del LED, nel caso in cui il valore acquisito sia all'interno dei limiti dell'allarme, utilizzeremo una subroutine, ma ne prepareremo anche un'altra per attivare l'uscita che mette in marcia il motore, e un'altra per configurare la trasmissione tramite USART.

Ricordatevi di definire le variabili necessarie all'inizio del programma.

All'interno del programma principale dovremo inviare i caratteri da visualizzare sul PC, quali la lettura della temperatura e il numero di pezzi prodotti. Eseguiremo una chiamata alla subroutine di trasmissione per ogni carattere. Per trasmettere i dati delle variabili è necessario convertirne il valore da binario a decimale, a questo scopo creeremo una subroutine per eseguire questa operazione.

Potete seguire il codice tramite le immagini di queste pagine oppure utilizzare il file che si trova sul secondo CD-R fornito con l'opera (Forno 75-125.asm). Ricordatevi di dichiarare sempre le variabili che mano a mano si vanno generando. Per i pezzi di tipo B cambierà solamente il valore delle costanti di temperatura, invece di 75 e 125 dovranno essere 100 e 150.

```
CICLO   clrwdt
        movlw   '\r'
        call    VISUAL
        movlw   'T'
        call    VISUAL
        movlw   'e'
        call    VISUAL
        movlw   'm'
        call    VISUAL
        movlw   'p'
        call    VISUAL
        movlw   ':'
        call    VISUAL
        movlw   ' '
        call    VISUAL

        movfw   ADRESH
        movwf   aux11
        call    bin2dec
        movfw   aux21
        addlw   '0'
        call    VISUAL
        movfw   aux31
        addlw   '0'
        call    VISUAL
        movfw   aux41
        addlw   '0'
        call    VISUAL
        clrwdt
        movlw   ' '
        call    VISUAL
        movlw   ' '
        call    VISUAL
        movlw   'P'
        call    VISUAL
        movlw   'e'
        call    VISUAL
        movlw   'z'
        call    VISUAL
        movlw   'z'
        call    VISUAL
        movlw   'i'
        call    VISUAL
        movlw   ':'
        call    VISUAL
        movlw   ' '
        call    VISUAL
```

```
        movfw   n_pezzi
        movwf   aux11
        call    bin2dec
        movfw   aux21
        addlw   '0'
        call    VISUAL
        movfw   aux31
        addlw   '0'
        call    VISUAL
        movfw   aux41
        addlw   '0'
        call    VISUAL
        movlw   .10
        movwf   aux51
tdesp   movlw   ' '
        call    VISUAL
        decfsz  aux51,F
        goto    tdesp
        banksel PORTB
        goto    eti
```

Programma principale.

```
;****************************************  conversione da bin a dec  ****************
bin2dec movwf   aux11
        clrf    aux21

        goto    salto
da100   incf    aux21
        movlw   .100
        subwf   aux11
salto   btfsc   aux11,7
        goto    da100
        btfss   aux11,6
        goto    no100
        btfss   aux11,5
        goto    no100
        btfsc   aux11,4
        goto    da100
        btfsc   aux11,3
        goto    da100
        btfsc   aux11,2
        goto    da100
no100   clrf    aux31
        goto    salto1
da10    incf    aux31
        movlw   .10
        subwf   aux11
salto1  btfsc   aux11,6
        goto    da10
        btfsc   aux11,5
        goto    da10
        btfsc   aux11,4
        goto    da10
        btfss   aux11,3
        goto    no10
        btfsc   aux11,2
        goto    da10
        btfsc   aux11,1
        goto    da10
no10    movfw   aux11
        movwf   aux41
        return


        end
```

Subroutine per convertire un numero binario in decimale.